AMENDMENT TO THE CLAIMS

1. (Currently Amended) An apparatus for cache flushing, comprising:

a list structure to track a status of a plurality of cache entries containing one bit per a variable number of cache lines, wherein said list structure is located outside a ~~cache and~~ cache, wherein said list structure does not contain cache data or ~~addresses~~ addresses, and wherein a logical arrangement of said list structure conforms to said variable number;

a query mechanism to check said list structure for the state of a cache entry; and

a cache flush mechanism, logically coupled to said list structure and the cache, to flush a cache entry and for modifying said list structure to reflect a flushed state.

2. (Original) An apparatus in accordance with claim 1, wherein:

said list structure comprises one bit per cache line.

3-5 (Cancelled)

6. (Currently Amended) An apparatus in accordance with ~~claim 5,~~ claim 1, wherein:

said variable number is set by an operating system.

7. (Previously Presented) An apparatus in accordance with claim 1, wherein:

a logical arrangement of said list structure matches an architecture of a cache.

8. (Previously Presented) An apparatus in accordance with claim 1, wherein:

said cache flush mechanism modifies a cache state responsive to results of a query of the said list structure.

9. (Original) An apparatus in accordance with claim 8, wherein:

said cache flush mechanism is logically coupled to a higher level cache for writing back modified data.

10. (Original) An apparatus in accordance with claim 8, wherein:

said cache flush mechanism based on the said list structure is logically coupled to a higher level cache for evicting modified data.

11. (Previously Presented) An apparatus in accordance with claim 8, wherein:

said cache flush mechanism is logically coupled to a main memory for writing back modified data.

12. (Previously Presented) An apparatus in accordance with claim 8, wherein:

said cache flush mechanism is logically coupled to a main memory for evicting modified data.

13. (Original) An apparatus in accordance with claim 1, wherein:

said list structure is located in random access memory (RAM).

14. (Previously Presented) An apparatus in accordance with claim 1, wherein:

said list structure is located on a die.

15. (Previously Presented) An apparatus in accordance with claim 1, further comprising:

a snoop command interpreter to check said list structure in response to a snoop command.

16. (Currently Amended) In a computer system with a cache memory, an apparatus for flushing the cache, comprising:

a list structure to record modifications to a plurality of cache entries ~~wherein~~ and containing one bit per a variable number of cache lines, wherein said list structure is located outside a cache and said list structures does not contain cache data or addresses and wherein a logical arrangement of said list structure conforms to said variable number;

a cache controller to query said list structure for modifications to said plurality of cache entries and generate a list of cache write-back instructions; and

wherein said cache controller is to invalidate said plurality of cache entries corresponding to said list of cache write-back instructions.

17. (Original) An apparatus in accordance with claim 16, wherein:

said list structure is a full list.

18. (Original) An apparatus in accordance with claim 16, wherein:

said list structure is a partial list.

19. (Previously Presented) An apparatus in accordance with claim 17, wherein:

said full list comprises one entry per cache line.

20. (Original) An apparatus in accordance with claim 18, wherein:

said partial list comprises one entry per plurality of cache lines.

21. (Previously Presented) In a multiprocessor computer system with a plurality of processors and cache memory, an apparatus for cache flushing, comprising:

a list structure to track a status of a plurality of cache entries, wherein said list structure is located outside a cache and wherein said list structure does not contain cache data or addresses;

a processor identification within said list structure to link each of said plurality of cache entries to one of the plurality of processors;

a query mechanism to check said list structure for a state of a cache entry identified with a processor;

a cache flush mechanism to flush a cache entry linked to an identified processor and to modify said list structure to reflect a flushed status.

22. (Original) An apparatus in accordance with claim 21, wherein:

said list structure contains at least one bit for each cache line.

23. (Original) An apparatus in accordance with claim 21, wherein:

said list structure contains at least one bit for each of a plurality of cache lines.

24. (Original) An apparatus in accordance with claim 21, wherein:

said list structure is located on a die with at least one of the plurality of processors.

25. (Currently Amended) A method of flushing a cache, comprising:

creating a table of cache entries separate from the cache and without the cache data or addresses containing one bit per a variable number of cache lines and wherein a logical arrangement of said table conforms to said variable number;

tracking modified cache entries in said table; and

generating a write-back command from said table in response to a cache flush event.

26. (Original) A method in accordance with claim 25, further comprising:

generating an invalidate command in response to a cache flush event.

27. (Previously Presented) A method in accordance with claim 25, further comprising: repeating the method for each level of cache.

28. (Original) A method in accordance with claim 25, further comprising: querying said table in response to a snoop command.

29. (Original) A method in accordance with claim 25, further comprising writing-back modified cache entries to memory.

30. (Original) A method in accordance with claim 25, further comprising: writing-back modified cache entries to a high level cache.